

## 修 士 論 文 概 要 書

2010 年 2 月提出

専攻名	情報理工学	氏名	露崎 浩太	指 導	上田 和紀 印
研究指導名	並列知識情報処理	学籍番号	5108B085 - 3 <sup>CD</sup>	教 員	
研 究 題 目	クラスタ向けハイブリッド SAT ソルバの設計と実装				

## 1 はじめに

ある命題論理式に対して、解となる変数の割り当てを示すか、解が存在しないことを示す問題を充足可能性問題 (SAT) と呼ぶ。SAT は CAD や検証技術のエンジンとして利用可能なため並列化を行い高速化する意義は大きい。しかし、SAT は NP 完全問題なため、解の探索に、最悪、指数オーダーの時間がかかる。さらに、解探索域の静的分割が困難なことから、より効果の高い並列化手法が研究されており、近年では探索域の動的分割や探索域削減に有益な学習節の共有といった手法が主流となっている。特に有益な学習節の効率的な共有は解探索の高速化に非常に効果が高いとされ、効率的な共有手法が研究されている。

本研究では MPI を用いてクラスタ向けに作られた学習節共有による並列効果の高い並列 SAT ソルバ、c-sat をベースに SAT Competition2009 で最も優秀な成績を修めた逐次ソルバである Precosat を並列化した c-preco、および c-preco と c-sat のワーカを融合したハイブリッド SAT ソルバである c-satw を実装し、SAT Competition2009 の問題においてクラスタ上での 31PE 実行で、Precosat より最大で約 62 倍、幾何平均で約 3 倍の性能を示すことができた。

## 2 SAT

SAT を高速に解くためのソフトウェアである SAT ソルバの多くは DPLL アルゴリズムに基づいており、変数選択、推論、衝突解析、バックトラックの 4 つのフェイズを繰り返し、解を探索する。解の探索木は巨大なうえ、問題やストラテジにより大きく異なるが、衝突解析フェイズで得られる、特定の探索域に解が存在しないことを示す学習節 (learned clause, lemma) を用いることにより、大きく剪定することが可能である。

## 2.1 c-sat

c-sat[2] は、古典的な DPLL SAT ソルバである MiniSat を MPI で並列に実行し、lemma の共有と探索域の動的分割によって探索域の大幅な剪定と高速な動作を実現した並列 SAT ソルバである。SAT ソルバでは並列化における通がボトルネックとなりやすいため、c-sat では冗長な lemma は共有しないといったルールの適用等によって通信量の削

減を図っている。また、通信による解探索効率低下を防ぐため、lemma の共有には専用のマスタを用いたマスタワーカ構造を用いている。

## 2.2 Precosat

Precosat[3] は SAT ソルバの競技会である SAT Competition2009 の Application 分野にて最も優秀な成績を収めた DPLL ベースの逐次 SAT ソルバである。Precosat では従来の DPLL アルゴリズムに加え、restart のタイミングに有効とされる Luby strategy[1] を採用している。また、lemma が 2 項であった際には、binary clause reasoning を用いて割り当てと制約伝播を優先的に行うことで、より高速な解の探索が可能となっている。

## 3 ハイブリッド SAT ソルバ

本研究では、まず c-sat のワーカを Precosat に組み替えた並列 Precosat である c-preco を実装し、次に従来のワーカである MiniSat と連動させた c-satw を実装した。

## 3.1 c-preco

c-preco は図 1 に示すように lemma の共有効率のよい c-sat をベースに、ワーカ部分の SAT ソルバを MiniSat より性能のよい Precosat に入れ替え、c-sat 全体の速度向上を目指したソルバである。また、最新の SAT ソルバでも c-sat 同様の lemma の共有効果が得られること確認するため、lemma 共有ルール、マスタワーカ構造などは c-sat のそれをそのまま引き継ぎ、さらに c-sat で実装されていた探索域の動的分割機能は停止させてある。

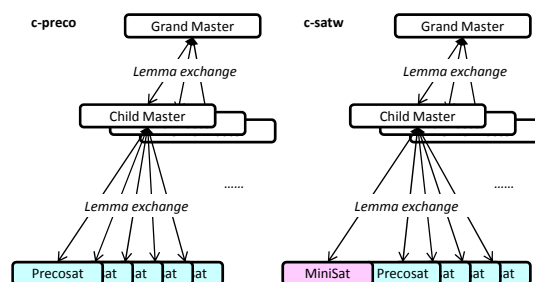


図 1 c-preco と c-satw の 3 層マスタワーカ構造

## 3.2 c-satw

c-satw では c-preco でワーカとして動作しているソルバのうち各子マスタにつながる 1 台ずつを従来の MiniSat に

戻し、Precosat の苦手とする問題への補助，ならびに Precosat とは全く異なる探索ストラテジから得られる lemma によって，さらに効率的な探索域の削減を目指したソルバである．lemma は元の問題の resolution によって得られるため，異なるソルバ間で利用可能ではあるが Precosat と MiniSat では実装手法が異なるため，lemma を正しく認識させる工夫として認識に関する情報を通信に加えるなどの実装を行った．

## 4 評価実験

今回は実験環境として AMD Dual-core Opteron 3.0GHz × 8 の計算機を 2 ノードつないだクラスタを利用した．各ノードのメモリは 128GB であり，通信には 10Gbps の Giga bit ethernet を利用し，通信ミドルウェアは MPICH2 を使用した．また，本実験では最新の大会である SAT Competition 2009 の本戦で利用された Application 分野の問題 292 問を利用した．大会同様，解探索の制限時間を 1200 秒に制限し，1200 秒かかって解けなかった問題は 1200 秒で解けたものとして，速度向上比を計算している．この結果を c-preco と c-satw について表 1，表 2 に示す．各表において，SAT は充足可能，UNS は充足不可能，UNK は解が元々示されていない問題を示し，数字は解けた問題数，() 内の数値は総問題数，[] 内の問題は該当問題を解くのにかった時間の総和を表している．また表中では c-preco を CPR，c-satw を CW と略記した．

表 1 c-preco の実行結果

	Precosat	CPR(7PE)	CPR(31PE)
SAT(99)	66[50457]	78[42499]	86[35185]
UNS(178)	100[115877]	111[105634]	125[97660]
UNK(15)	0[18000]	0[18000]	0[18000]
total	166[184335]	189[166133]	211[152155]

表 2 c-satw の実行結果

	Precosat	CW(7PE)	CW(31PE)
SAT(99)	66[50457]	82[36387]	91[26927]
UNS(178)	100[115877]	114[102814]	125[89882]
UNK(15)	0[18000]	0[18000]	0[18000]
total	166[184335]	196[152155]	216[134810]

次に各ソルバ 31PE 実行時の SAT と UNS における性能向上比をそれぞれプロットしたものを図 2 に示す．向上比の対比をとるためプロットは両ソルバが 1200 秒以内に解け，Precosat で 60 秒以上かかる SAT44 題，UNS61 題について行った．この条件に当てはまらないため UNK 問題のプロットは省略した．この問題の 31PE 時の性能向上比の幾何平均と最大値を表 3 に示す．| で区切られた数字のうち左が幾何平均，右が最大値を表す．

表 3 c-preco と c-satw の性能向上比 (幾何平均 | 最大)

	CPR(31PE)	CW(31PE)
SAT(44)	2.98   29.41	4.07   62.05
UNS(61)	2.28   4.83	2.60   8.68
total	2.55	3.14

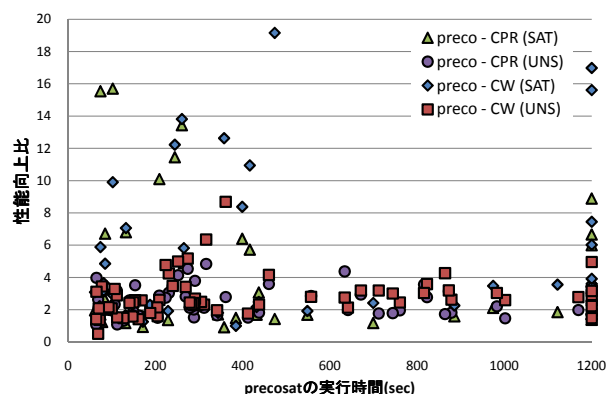


図 2 各問題における c-preco と c-satw の性能向上比

## 5 まとめと考察

今回の実験より，c-preco，c-satw 共に，Precosat より性能を向上し，並列化の効果が確認できた．コンペティションとして最新であり難しい問題が用意された SAT2009 の問題であっても多くの問題が 2～20 倍程度の速度向上を得られており，また，UNS より SAT のほう大きく性能向上が得られていることがわかる．特に解ける問題数としては総計で 50 問以上増え，31PE で約 3 倍，SAT 問題で最大で 62 倍の性能向上が得られた．また，c-satw では c-sat，c-preco，Precosat のどのソルバでも解くことのできなかった問題を SAT6 題，UNS1 題の計 7 題多く解くことができ，逐次としては遅いソルバであってもハイブリッド化として並列ソルバへの導入する意義は十分にあったと言える．しかし，全体的に性能向上しているものの元々 Precosat が求解に 500～1200 秒かかっている問題の性能向上比はせいぜい約 5 倍程度にとどまった．求解に時間のかかる問題においては通信の回数，lemma の量が増え，それらが性能に影響してくることが十分に考えられるため，従来の c-sat より精度のよい lemma 選定ストラテジのチューニングが今後の課題である．

## 参考文献

- [1] Luby, M., Sinclair, A., Zuckerman, D., Optimal Speedup of Las Vegas Algorithms, *Information Processing Letters*, Vol. 47, pp. 173–180, 1993.
- [2] Ohmura, K., Ueda, K., c-sat: A Parallel SAT Solver for Clusters, In *Theory and Applications of Satisfiability Testing*, pp. 524–537. Springer Verlag, 2009.
- [3] Biere, A., Pre,icoSAT@SC'09, In *SAT 2009 competitive events booklet*, pp. 41–43, 2009.